

# Referencia Rápida de find

Búsqueda de archivos por nombre, tipo, tamaño, tiempo, permisos y acciones

## Búsqueda básica

### Ejecutar find

```
find . # listar todos los archivos
recursivamente
find /var/log # buscar desde ruta específica
find . -name "*.txt" # buscar por nombre
find / -name "config" 2>/dev/null # suprimir errores de permisos
find dir1 dir2 -name "*.py" # buscar en múltiples directorios
```

### Resumen de sintaxis

<b>find</b> [path... ] [tests] [actions]	Forma general — rutas, luego tests, luego acciones
<b>path</b>	Directorio de inicio (por defecto: directorio actual)
<b>test</b>	Condición para filtrar archivos (-name, -type, etc.)
<b>action</b>	Qué hacer con los resultados (-print, -exec, etc.)
<b>Acción por defecto</b>	-print si no se especifica acción

## Patrones de nombre

### Coincidencia de nombre

```
find . -name "*.log" # glob con distinción de mayúsculas
find . -iname "readme*" # glob sin distinción de mayúsculas
find . -name "*.py" -o -name "*.js" # 0: archivos Python o JS
find . -path "*/src/*.ts" # coincidir con ruta completa
find . -regex '.*\.(py|js)\.'
```

### Tests de nombre

<b>-name pattern</b>	Nombre base coincide con glob (distingue mayúsculas)
<b>-iname pattern</b>	Nombre base coincide con glob (sin distinguir)
<b>-path pattern</b>	Ruta completa coincide con glob
<b>-ipath pattern</b>	Ruta completa coincide con glob (sin distinguir)
<b>-regex pattern</b>	Ruta completa coincide con expresión regular
<b>-iregex pattern</b>	Ruta completa coincide con regex (sin distinguir)

## Filtros de tipo

### Filtrar por tipo

```
find . -type f # solo archivos regulares
find . -type d # solo directorios
find . -type l # enlaces simbólicos
find . -type f -name "*.sh" # combinar tipo + nombre
```

### Tipos de archivo

<b>-type f</b>	Archivo regular
<b>-type d</b>	Directorio
<b>-type l</b>	Enlace simbólico
<b>-type b</b>	Dispositivo de bloque
<b>-type c</b>	Dispositivo de caracteres
<b>-type p</b>	Tubería con nombre (FIFO)
<b>-type s</b>	Socket
<b>-empty</b>	Archivo o directorio vacío

## Tamaño y tiempo

### Ejemplos de tamaño y tiempo

```
find . -size +100M # mayor de 100 MB
find . -size -1k # menor de 1 KB
find . -mtime -7 días # modificado en los últimos 7 días
find . -mmin -30 # modificado en los últimos 30 minutos
find . -newer reference.txt # más reciente que el archivo de referencia
```

### Tests de tamaño y tiempo

<b>-size +/-Nc</b>	Tamaño en bytes (c), kilobytes (k), megabytes (M), gigabytes (G)
<b>-mtime +/-N</b>	Modificado hace N*24 horas (+más antiguo, -más reciente)
<b>-atime +/-N</b>	Accedido hace N*24 horas
<b>-ctime +/-N</b>	Estado cambiado hace N*24 horas
<b>-mmin +/-N</b>	Modificado hace N minutos
<b>-newer file</b>	Modificado más recientemente que el archivo
<b>-newermt date</b>	Modificado después de la cadena de fecha (GNU)

## Permisos

### Ejemplos de permisos

```
find . -perm 644 # permisos exactos: rw-r--r--
find . -perm -u+x # usuario tiene bit de ejecución
find . -perm /o+w # otros tienen escritura (cualquier coincidencia)
find . -user root # propiedad de root
find . -group www-data -type f # propiedad del grupo
```

### Tests de permisos

<b>-perm mode</b>	Coincidencia exacta de permisos
<b>-perm -mode</b>	Todos los bits especificados están activos
<b>-perm /mode</b>	Cualquier bit especificado está activo
<b>-user name</b>	Propiedad del usuario (nombre o UID)
<b>-group name</b>	Propiedad del grupo (nombre o GID)
<b>-nouser</b>	Sin usuario coincidente en /etc/passwd
<b>-nogroup</b>	Sin grupo coincidente en /etc/group

## Acciones

### Ejemplos de acciones

```
find . -name "*.log" -print # imprimir rutas (por defecto)
find . -name "*.tmp" -delete # eliminar archivos coincidentes
find . -type f -ls # listado detallado
find . -name "*.txt" -print0 # salida delimitada por nulo
find . -type f -printf "%p %s\n" # formato personalizado (GNU)
```

### Referencia de acciones

<b>-print</b>	Imprimir ruta (delimitado por salto de línea)
<b>-print0</b>	Imprimir ruta (delimitado por nulo, seguro para xargs)
<b>-ls</b>	Imprimir detalles del archivo (como <b>ls -dils</b> )
<b>-delete</b>	Eliminar archivos coincidentes (implica <b>-depth</b> )
<b>-printf format</b>	Formato de salida personalizado (GNU): %p ruta, %s tamaño, %t tiempo
<b>-fprint file</b>	Escribir rutas en archivo
<b>-quit</b>	Salir después de la primera coincidencia

## Combinar tests

### Operadores lógicos

```
find . -name "*.py" -type f # Y implícito
find . -name "*.py" -a -size +10k # Y explícito
find . -name "*.py" -o -name "*.js" # 0
find . ! -name "*.log" # NO
find . \( -name "*.py" -o -name "*.js" \) -type f
```

### Referencia de operadores

<b>expr1 expr2 / expr1 -a expr2</b>	Y — ambas deben ser verdaderas (por defecto)
<b>expr1 -o expr2</b>	O — cualquiera debe ser verdadera
<b>! expr / -not expr</b>	NO — negar la expresión
<b>\( expr \)</b>	Agrupar expresiones (escapar paréntesis en shell)
<b>Orden de evaluación</b>	De izquierda a derecha; <b>-a</b> tiene mayor precedencia que <b>-o</b>

## Exec y eliminación

### Ejemplos de exec

```
find . -name "*.sh" -exec chmod +x {} \;
find . -name "*.log" -exec rm {} +
find . -type f -exec grep -l "TODO" {} +
find . -name "*.bak" -ok rm {} \; # confirmar antes de cada uno
find . -name "*.tmp" -print0 | xargs -0 rm
```

### Referencia de exec

<b>-exec cmd {} \;</b>	Ejecutar cmd una vez por archivo ({} = ruta del archivo)
<b>-exec cmd {} +</b>	Ejecutar cmd con múltiples archivos a la vez (más rápido)
<b>-ok cmd {} \;</b>	Como <b>-exec</b> pero pide confirmación
<b>-execdir cmd {} \;</b>	Ejecutar cmd desde el directorio del archivo
<b>xargs -0</b>	Combinar con <b>-print0</b> para procesamiento seguro en lote
<b>-delete</b>	Eliminar archivos; procesar primero los más profundos

## Profundidad y poda

### Ejemplos de profundidad y poda

```
find . -maxdepth 1 -type f # solo directorio actual
find . -mindepth 2 -name "*.py" # omitir nivel superior
find . -name ".git" -prune -o -print # omitir dirs .git
find . -depth -name "*.tmp" -delete # procesar hijos primero
```

### Opciones de profundidad

<b>-maxdepth N</b>	Descender como máximo N niveles (0 = solo ruta inicial)
<b>-mindepth N</b>	No aplicar tests en niveles menores que N
<b>-depth</b>	Procesar contenidos del directorio antes que el propio directorio
<b>-prune</b>	No descender en el directorio coincidente
<b>-mount / -xdev</b>	No cruzar límites de sistemas de archivos
<b>-follow / -L</b>	Seguir enlaces simbólicos

## Patrones comunes

### One-liners

```
find . -name "*.pyc" -delete # limpiar bytecode Python
find . -type f -size 0 -delete # eliminar archivos vacíos
find . -mtime +30 -name "*.log" -delete # purgar logs antiguos
find . -type f -name "*.md" | wc -l # contar archivos Markdown
find . -type d -empty -delete # eliminar dirs vacíos
```

## Referencia Rápida de find

---

### Recetas

Encontrar archivos más grandes	<pre>find . -type f - printf '%s %p\n'   sort -rn   head</pre>
Encontrar duplicados por nombre	<pre>find . -type f   awk -F/ '{print \$NF}'   sort   uniq -d</pre>
Renombrar extensión	<pre>find . -name  '*.txt' -exec rename  's/.txt/.md/' {} +</pre>
Encontrar enlaces simbólicos rotos	<pre>find . -xtype l</pre>
Archivar archivos recientes	<pre>find . -mtime -7 - print0   tar czf recent.tar.gz -- null -T -</pre>
Buscar en archivos de código	<pre>find . -name '*.py' -exec grep -l 'pattern' {} +</pre>